

1 **CLAIMS**

2

3 1. A method, comprising:

4 generating an image of an operating system with a host computing device;

5 communicating the image of the operating system to a software
6 development peripheral;

7 executing the operating system corresponding to the image with the
8 software development peripheral;

9 communicating information generated by the operating system to the host
10 computing device; and

11 displaying the information generated by the operating system with the host
12 computing device.

13

14 2. A method as recited in claim 1, further comprising recognizing a
15 configuration identification of the software development peripheral with a cross-
16 platform development component of the host computing device.

17

18 3. A method as recited in claim 1, wherein generating includes
19 generating the image of the operating system with a cross-platform development
20 component of the host computing device.

1 4. A method as recited in claim 1, further comprising recognizing a
2 configuration identification of the software development peripheral with a cross-
3 platform development component of the host computing device, and wherein
4 generating includes generating the image of the operating system with the cross-
5 platform development component, the image of the operating system
6 corresponding to the configuration identification of the software development
7 peripheral.

8

9 5. A method as recited in claim 1, further comprising debugging the
10 information generated by the operating system with a cross-platform development
11 component of the host computing device.

12

13 6. A method as recited in claim 1, further comprising connecting the
14 software development peripheral to a network via a network communication driver
15 of the host computing device, the network communication driver communicatively
16 linked with the network and with a virtual network communication driver of the
17 software development peripheral.

18

19 7. A method as recited in claim 1, wherein communicating includes
20 communicating the information generated by the operating system to the host
21 computing device via a debug transport.

1 **8.** A method as recited in claim 1, wherein communicating includes
2 communicating the information generated by the operating system to the host
3 computing device with a virtual device driver of the software development
4 peripheral.

5
6 **9.** A method as recited in claim 1, wherein communicating includes
7 communicating image data generated by the operating system to a virtual
8 input/output system of the host computing device with a virtual device driver of
9 the software development peripheral.

10
11 **10.** A method as recited in claim 1, further comprising receiving a
12 keyboard input with the software development peripheral from a virtual
13 input/output system of the host computing device, the keyboard input generated
14 with a keyboard connected to the host computing device.

15
16 **11.** A method as recited in claim 1, further comprising receiving a
17 pointing device input with the software development peripheral from a virtual
18 input/output system of the host computing device, the pointing device input
19 generated with a pointing device connected to the host computing device.

1 **12.** A software development peripheral performing a method,
2 comprising:
3

4 providing a configuration identification of the software development
5 peripheral to an operating system development component of a host computing
6 device;

7 receiving an image of an operating system, the image of the operating
8 system generated with the operating system development component;

9 executing the operating system corresponding to the image; and

10 communicating information generated by the operating system to the
11 operating system development component.

12 **13.** A method as recited in claim 12, wherein receiving includes
13 receiving an image of the operating system that corresponds to the configuration
14 identification of the software development peripheral.

15
16 **14.** A method as recited in claim 12, further comprising
17 communicatively linking to a network with a virtual network communication
18 driver, the virtual network communication driver communicatively linked with a
19 network communication driver of the host computing device.

20
21 **15.** A method as recited in claim 12, wherein communicating includes
22 communicating the information generated by the operating system to the host
23 computing device with a virtual device driver via a debug transport.

1 **16.** A method as recited in claim 12, wherein communicating includes
2 communicating the information generated by the operating system to a virtual
3 input/output system of the host computing device with a virtual device driver.

4

5 **17.** A method as recited in claim 12, further comprising communicating
6 image data generated by the operating system to the host computing device for
7 display.

8

9 **18.** A method as recited in claim 12, further comprising communicating
10 image data generated by the operating system to a virtual input/output system of
11 the host computing device with a virtual display device driver for display at the
12 host computing device.

13

14 **19.** A method as recited in claim 12, further comprising receiving a
15 keyboard input that is generated with a keyboard connected to the host computing
16 device.

17

18 **20.** A method as recited in claim 12, further comprising receiving a
19 pointing device input that is generated with a pointing device connected to the host
20 computing device.

1 **21.** One or more computer-readable media comprising computer
2 executable instructions that, when executed, direct a software development
3 peripheral to perform a method comprising:

4 receiving an image of an operating system from a host computing device,
5 the image of the operating system corresponding to a configuration identification
6 of the software development peripheral;

7 executing the operating system corresponding to the image; and

8 communicating information generated by the operating system to an
9 operating system development component of the host computing device.

10
11 **22.** One or more computer-readable media as recited in claim 21,
12 wherein communicating includes communicating the information generated by the
13 operating system to the operating system development component via a debug
14 transport.

15
16 **23.** One or more computer-readable media as recited in claim 21,
17 wherein the method further comprises communicating peripheral device output
18 information generated by the operating system to a virtual input/output system of
19 the host computing device with a virtual device driver.

20
21 **24.** One or more computer-readable media as recited in claim 21,
22 wherein the method further comprises communicating image data generated by the
23 operating system to a virtual input/output system of the host computing device for
24 display.

1 **25.** One or more computer-readable media as recited in claim 21,
2 wherein the method further comprises communicating image data generated by the
3 operating system to a virtual input/output system of the host computing device
4 with a virtual display device driver.

5
6 **26.** A system, comprising:
7 a host computing device configured to generate an image of an operating
8 system; and

9 a software development peripheral configured to:
10 receive the image of the operating system from the host computing
11 device;
12 execute the operating system corresponding to the image; and
13 communicate information generated by the operating system to the host
14 computing device for display.

15
16 **27.** A system as recited in claim 26, wherein the host computing device
17 includes a first type of processor to generate the image of the operating system,
18 and wherein the software development peripheral is configured to execute the
19 operating system on a second type of processor, the second type of processor
20 being different than the first type of processor.

1 **28.** A system as recited in claim 26, wherein the host computing device
2 is further configured to recognize the software development peripheral as a plug
3 and play device when the software development peripheral is communicatively
4 linked with the host computing device.

5

6 **29.** A system as recited in claim 26, wherein the host computing device
7 includes a cross-platform development component configured to recognize a
8 configuration identification of the software development peripheral when the
9 software development peripheral is communicatively linked with the host
10 computing device.

11

12 **30.** A system as recited in claim 26, wherein the host computing device
13 includes a cross-platform development component configured to generate the
14 image of the operating system.

15

16 **31.** A system as recited in claim 26, wherein the host computing device
17 includes a cross-platform development component configured to recognize a
18 configuration identification of the software development peripheral when the
19 software development peripheral is communicatively linked with the host
20 computing device, and wherein the cross-platform development component is
21 further configured to generate the image of the operating system corresponding to
22 the configuration identification of the software development peripheral.

1 **32.** A system as recited in claim 26, wherein the host computing device
2 includes a cross-platform development component configured to debug the
3 information generated by the operating system.

4

5 **33.** A system as recited in claim 26, wherein the host computing device
6 and the software development peripheral are communicatively linked via a debug
7 transport.

8

9 **34.** A system as recited in claim 26, wherein the host computing device
10 and the software development peripheral are communicatively linked via a
11 universal serial bus connection.

12

13 **35.** A system as recited in claim 26, wherein the software development
14 peripheral includes a virtual device driver configured to route the information
15 generated by the operating system to the host computing device, and wherein the
16 host computing device includes a virtual input/output system configured to receive
17 the information generated by the operating system.

18

19 **36.** A system as recited in claim 26, wherein the host computing device
20 includes a virtual input/output system configured to receive the information
21 generated by the operating system and route the information to a display device.

22

23 **37.** A system as recited in claim 26, wherein the software development
24 peripheral is further configured to communicate image data generated by the
25 operating system to the host computing device via a virtual display device driver.

1
2 **38.** A system as recited in claim 26, wherein the software development
3 peripheral is further configured to communicate image data generated by the
4 operating system to the host computing device via a virtual display device driver,
5 and wherein the host computing device includes a virtual input/output system
6 configured to receive the image data and route the image data to a display device.

7
8 **39.** A system as recited in claim 26, wherein the software development
9 peripheral is further configured to connect to a network via a network
10 communication driver of the host computing device, the network communication
11 driver communicatively linked with the network and with a virtual network
12 communication driver of the software development peripheral.

13
14 **40.** A system as recited in claim 26, wherein the host computing device
15 includes a virtual input/output system configured to route a keyboard input to the
16 software development peripheral.

17
18 **41.** A system as recited in claim 26, wherein the host computing device
19 includes a virtual input/output system configured to route a pointing device input
20 to the software development peripheral.

1 **42.** A software development peripheral, comprising:
2 a memory component configured to maintain an image of an operating
3 system received from a host computing device;
4 a processor configured to execute the operating system corresponding to the
5 image; and
6 a virtual device driver configured to communicate information generated by
7 the operating system to the host computing device.

8

9 **43.** A software development peripheral as recited in claim 42, further
10 comprising a configuration identification to identify the software development
11 peripheral to the host computing device when the software development peripheral
12 is communicatively linked with the host computing device.

13

14 **44.** A software development peripheral as recited in claim 42, further
15 comprising a configuration identification to identify the software development
16 peripheral to a cross-platform development component of the host computing
17 device when the software development peripheral is communicatively linked with
18 the host computing device.

19

20 **45.** A software development peripheral as recited in claim 42, wherein
21 the virtual device driver communicates the information generated by the operating
22 system to the host computing device for display.

1 **46.** A software development peripheral as recited in claim 42, wherein
2 the virtual device driver is a virtual display device driver configured to
3 communicate image data generated by the operating system to the host computing
4 device for display.

5

6 **47.** A software development peripheral as recited in claim 42, wherein
7 the virtual device driver communicates the information generated by the operating
8 system to be debugged at the host computing device.

9

10 **48.** A software development peripheral as recited in claim 42, wherein
11 the virtual device driver communicates the information generated by the operating
12 system to the host computing device via a debug transport.

13

14 **49.** A software development peripheral as recited in claim 42, wherein
15 the virtual device driver communicates the information generated by the operating
16 system to the host computing device via a universal serial bus connection.

17

18 **50.** A software development peripheral as recited in claim 42, wherein
19 the virtual device driver communicates the information generated by the operating
20 system to a virtual input/output system of the host computing device.

1 **51.** A software development peripheral as recited in claim 42, further
2 comprising a virtual network communication driver configured to connect to a
3 network via a network communication driver of the host computing device, the
4 network communication driver communicatively linked with the network.

5
6 **52.** A software development peripheral as recited in claim 42, wherein
7 the processor is further configured to receive input from a keyboard that is
8 connected to the host computing device.

9
10 **53.** A software development peripheral as recited in claim 42, wherein
11 the processor is further configured to receive input from a pointing device that is
12 connected to the host computing device.

13
14 **54.** A software development peripheral as recited in claim 42, further
15 comprising an expansion component configured to couple a peripheral device with
16 the software development peripheral.

17
18 **55.** A software development peripheral as recited in claim 42, further
19 comprising an expansion component configured to couple a display device with
20 the software development peripheral.

21
22 **56.** A software development peripheral as recited in claim 42, further
23 comprising an expansion component configured to couple an input device with the
24 software development peripheral.

1 **57.** A software development peripheral, comprising:
2 means for receiving an image of an operating system, the image of the
3 operating system generated with a host computing device;
4 means for executing the operating system corresponding to the image; and
5 means for communicating information generated by the operating system to
6 a virtual input/output system of the host computing device.

7
8 **58.** A software development peripheral as recited in claim 57, further
9 comprising means for communicating image data generated by the operating
10 system to the host computing device for display.

11
12 **59.** A software development peripheral as recited in claim 57, further
13 comprising means for receiving a keyboard input that is generated with a keyboard
14 connected to the host computing device.

15
16 **60.** A software development peripheral as recited in claim 57, further
17 comprising means for receiving a pointing device input that is generated with a
18 pointing device connected to the host computing device.